# Assignment 9-10　Integer Division 5

The problem is to compute the quotient $q$ and the remainder $r$ of a positive integer $a$ divided by a positive integer $b$. ($b \le a < 10^{250}$)

## Input

The input consists of $t$ ($30 \le t \le 40$) test cases. The first line of the input contains only positive integer $t$. Then $t$ test cases follow. Each test case consists of two lines which give the two positive integers $a$ and $b$ ($b \le a < 10^{250}$), respectively.

## Output

For each test case, you are to output exactly two lines containing, the quotient $q$ and the remainder $r$, respectively.

## Sample Input

```
3
12345
12311
12345
12345
12345
1
```

## Sample Output

```
1
34
1
0
12345
0
```

## Part of the program

You are required to write the function `division`, `subtraction` and `less` to complete the following program which solves this problem. In your program, you cannot declare global variables or static arrays.

```cpp
#include <iostream>
#include <cstring>
using std::cin;
using std::cout;
using std::endl;

#include <vector>
using std::vector;

// quotient = dividend / divisor; remainder = dividend % divisor
// provided that dividend != 0, divisor != 0 and dividend >= divisor
void division( vector< int > dividend, vector< int > divisor,
               vector< int > &quotient, vector< int > &remainder );

// hugeInt /= 10, or equivalently, shifts right by one position
void divideBy10( vector< int > &hugeInt );

// minuend -= subtrahend
// provided that minuend != 0, subtrahend != 0 and minuend >= subtrahend
void subtraction( vector< int > &minuend, vector< int > subtrahend );

// returns true if and only if hugeInt1 < hugeInt2
// provided that hugeInt1 != 0 and hugeInt2 != 0
bool less( vector< int > hugeInt1, vector< int > hugeInt2 );

// return true if and only if hugeInt1 == hugeInt2
// provided that hugeInt1 != 0 and hugeInt2 != 0
bool equal( vector< int > hugeInt1, vector< int > hugeInt2 );

// returns true if and only if the specified huge integer is zero
bool isZero( vector< int > hugeInt );

const int arraySize = 250;

int main()
{
   char strA[ 251 ], strB[ 251 ];

   int T;
   cin >> T;
   for( int t = 0; t < T; ++t )
   {
      cin >> strA >> strB;

      int dividendSize = strlen( strA );
      vector< int > dividend( dividendSize );
      for( int i = 0; i < dividendSize; ++i )
         dividend[ i ] = strA[ dividendSize - 1 - i ] - '0';

      int divisorSize = strlen( strB );
      vector< int > divisor( divisorSize );
      for( int i = 0; i < divisorSize; ++i )
         divisor[ i ] = strB[ divisorSize - 1 - i ] - '0';

      vector< int > quotient;
      vector< int > remainder;
      division( dividend, divisor, quotient, remainder );

      for( int i = quotient.size() - 1; i >= 0; i-- )
         cout << quotient[ i ];
      cout << endl;

      for( int i = remainder.size() - 1; i >= 0; i-- )
         cout << remainder[ i ];
      cout << endl;
   }
}
```

```
// quotient = dividend / divisor; remainder = dividend % divisor
// provided that dividend != 0, divisor != 0 and dividend >= divisor
void division( vector< int > dividend, vector< int > divisor,
               vector< int > &quotient, vector< int > &remainder )
{




}

// hugeInt /= 10, or equivalently, shifts right by one position
void divideBy10( vector< int > &hugeInt )
{
   int size = hugeInt.size();
   if( size == 1 )
      hugeInt[ 0 ] = 0;
   else
   {
      for( int i = 1; i < size; i++ )
         hugeInt[ i - 1 ] = hugeInt[ i ];
      hugeInt.pop_back();
   }
}

// minuend -= subtrahend
// provided that minuend != 0, subtrahend != 0 and minuend >= subtrahend
void subtraction( vector< int > &minuend, vector< int > subtrahend )
{




}

// returns true if and only if hugeInt1 < hugeInt2
// provided that hugeInt1 != 0 and hugeInt2 != 0
bool less( vector< int > hugeInt1, vector< int > hugeInt2 )
{




}

// return true if and only if hugeInt1 == hugeInt2
// provided that hugeInt1 != 0 and hugeInt2 != 0
bool equal( vector< int > hugeInt1, vector< int > hugeInt2 )
{
   int size1 = hugeInt1.size();
   int size2 = hugeInt2.size();
   if( size1 != size2 )
      return false;

   for( int i = size1 - 1; i >= 0; i-- )
      if( hugeInt1[ i ] != hugeInt2[ i ] )
         return false;

   return true;
}

// returns true if and only if the specified huge integer is zero
bool isZero( vector< int > hugeInt )
{
   return hugeInt.size() == 1 && hugeInt[ 0 ] == 0;
}
```